

PERBANDINGAN AKURASI ALGORITME PELATIHAN DALAM JARINGAN SYARAF TIRUAN UNTUK PERAMALAN JUMLAH PENGGUNA KERETA API DI PULAU JAWA

Budho Setyonugroho¹⁾, Adhistya Erna Permanasari²⁾, Sri Suning Kusumawardani³⁾

^{1), 2), 3)} Departemen Teknik Elektro dan Teknologi Informasi, Fakultas Teknik, Universitas Gadjah Mada

Jln. Grafika 2 Yogyakarta 55281 Indonesia

Email : budhosn@gmail.com¹⁾, adhistya@ugm.ac.id²⁾, suning@ugm.ac.id³⁾

Abstrak

Peramalan jumlah pengguna kereta api merupakan salah satu komponen penting dalam perencanaan untuk menentukan langkah-langkah strategis dalam suatu perusahaan. Jaringan syaraf tiruan merupakan salah satu metode yang telah banyak digunakan untuk menyelesaikan kasus peramalan. Kesulitan utama dalam implementasi metode jaringan syaraf tiruan dalam peramalan adalah menemukan kombinasi arsitektur yang tepat, menentukan nilai parameter laju pembelajaran yang sesuai dan pemilihan algoritme pelatihan yang optimal.

Dalam makalah ini akan melakukan simulasi dengan menerapkan algoritme bayesian regularization (*trainbr*), algoritme Scaled Conjugate Gradient (*trainscg*), algoritme levenberg-marquardt (*trainlm*) dengan variasi jumlah neuron pada lapisan tersembunyi dan laju pembelajaran yang berbeda untuk menemukan kombinasi arsitektur yang tepat untuk menyelesaikan kasus peramalan pengguna kereta api.

Dari simulasi yang dilakukan pada proses pelatihan yang menggunakan data runtun waktu dalam periode sepuluh tahun, arsitektur 12-30-1 dengan laju pembelajaran 0,5 dengan menggunakan algoritme pelatihan levenberg marquardt menunjukan hasil yang paling optimal dengan nilai MSE 0,000192 dan pada proses pengujian menunjukan nilai MAPE 23% yang termasuk dalam kategori cukup untuk digunakan dalam peramalan.

Kata kunci: jaringan syaraf tiruan, algoritme pelatihan, *trainlm*, *trainbr*, *trainscg*

1. Pendahuluan

Saat ini kereta api telah menjadi salah satu moda transportasi andalan masyarakat Indonesia untuk melakukan perjalanan jarak menengah dan jauh, khususnya di Pulau Jawa dan Sumatera. Data dari laporan tahunan PT KAI Indonesia tahun 2015, PT KAI telah mengangkut penumpang sebanyak 327,2 juta pada tahun 2015. Jumlah tersebut naik 16,73% dibandingkan tahun 2014 mencapai 280,3 juta penumpang [1]. Tingginya minat masyarakat terhadap angkutan kereta api, menjadi nilai tambah bagi perusahaan untuk dapat memberikan layanan yang optimal. Oleh karena itu, peramalan tentang jumlah pengguna kereta api menjadi hal yang penting bagi perusahaan karena dengan

mengetahui jumlah pengguna di masa yang akan datang.

Jaringan syaraf tiruan telah banyak dipakai untuk meramalkan apa yang terjadi di masa yang akan datang berdasarkan pola kejadian yang ada dimasa lampau [2]. Jaringan syaraf tiruan mengidentifikasi pola data dari sistem peramalan dapat dilakukan dengan menggunakan metode pendekatan pelatihan (*training*). Berdasarkan kemampuan belajar yang dimiliki oleh jaringan syaraf tiruan, maka pada tahap pelatihan akan mempelajari serta menganalisis pola data masa lalu dengan target atau keluaran yang diinginkan. Tahap pelatihan akan berusaha mencari suatu formula atau fungsi yang akan menghubungkan pola data masa lalu dengan keluaran yang diinginkan. Fungsi tersebut menggambarkan ketergantungan nilai data saat ini terhadap nilai data sebelumnya. Pada proses pelatihan, data historis dapat dipelajari dengan menyerap pengetahuan yang ada pada data dan direpresentasikan oleh nilai bobot yang akan memperkuat atau memperlemah sinyal antar koneksi yang dimiliki. Oleh karena itu diperlukan algoritme pelatihan dengan akurasi yang terbaik.

Pada penelitian ini akan menggunakan Jaringan syaraf tiruan dengan model *feed-forward backpropagation* dengan membandingkan akurasi dari fungsi algoritme pelatihan bayesian regularization (*trainbr*), algoritme Scaled Conjugate Gradient (*trainscg*), algoritme levenberg-marquardt (*trainlm*) untuk mengidentifikasi kesesuaian pola data input dan output dalam peramalan jumlah pengguna kereta api.

2. Landasan Teori

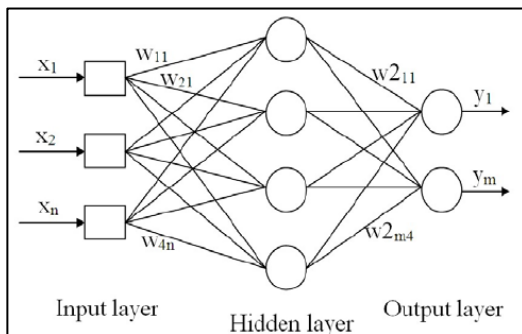
a) Runtun Waktu

Runtun waktu atau *time series* adalah data yang dikumpulkan berdasarkan interval waktu (minggu, bulan, tahun. Tujuan peramalan *time series* adalah memprediksi kejadian masa depan berdasarkan dari kejadian yang di amati saat ini [2].

b) Jaringan Syaraf Tiruan

Jaringan syaraf tiruan merupakan model simulasi sistem syaraf manusia melalui algoritme komputer [3]. Jaringan syaraf tiruan terdiri dari lapisan input, lapisan output, dan lapisan tersembunyi. Lapisan ini terdiri dari beberapa *neuron* yang berfungsi untuk menerima informasi dari luar, lapisan terakhir merupakan lapisan output untuk mendapatkan keluaran hasil pemrosesan jaringan, dan lapisan tersembunyi yang berada di antara lapisan input dan lapisan output. Setiap lapisan

memiliki *neuron* saling berhubungan dengan *neuron* lapisan lain. Model jaringan ini dikenal dengan model *multilayer* atau *feedforward neural network* [4]. Gambar 1 berikut merupakan contoh jaringan syaraf tiruan yang terdiri dari 3 lapisan.



Gambar 1 Jaringan Multilayer

c) Algoritme Pelatihan

Proses pelatihan merupakan salah satu parameter yang sangat penting dalam jaringan syaraf tiruan [5]. Algoritme atau fungsi pelatihan berfungsi untuk menentukan nilai bobot hubungan antar *neuron*.

Pada makalah ini, proses pelatihan akan menggunakan algoritme *bayesian regularization*, *quasi newton* dan *levenberg marquardt*.

Algoritme *bayesian regularization* adalah algoritme yang memperbaharui nilai bobot dan bias sesuai dengan optimasi *levenberg marquardt*. *Bayesian regularization* meminimalkan kuadrat kesalahan dan bobot, kemudian menentukan kombinasi yang benar sehingga dapat menggenarilasi suatu jaringan dengan baik.

Pada algoritme *levenberg marquardt*, perubahan nilai bobot dan bias sesuai dengan optimasi *levenberg marquardt*.

Sedangkan pada algoritme *quasi newton*, perubahan nilai bobot bias berdasarkan metode *newton bfgs* [6].

3. Metodologi

Pada makalah ini, membangun dan simulasi jaringan syaraf tiruan menggunakan perangkat lunak Matlab. Data yang digunakan adalah data jumlah pengguna kereta api di pulau Jawa pada periode 2006-2016 yang didapat dari situs Badan Pusat Statistik.

Sebelum dilakukan proses peramalan, data tersebut diolah dengan cara normalisasi yaitu mentransformasikan data menjadi interval [0.1, 0.9] agar mudah dalam proses pelatihan.

Langkah-langkah dalam pemilihan arsitektur optimal dilakukan dengan menentukan beberapa parameter berikut.

1) Membangun arsitektur jaringan

Arsitektur jaringan yang digunakan terdiri dari 3 lapisan yang terdiri dari 1 lapisan input, 1 lapisan tersembunyi dan 1 lapisan output.

Pada lapisan input, jumlah yang akan digunakan adalah 12 unit *neuron*. Jumlah *neuron* pada ini

berdasarkan jumlah data dalam periode satu tahun yaitu terdiri dari 12 bulan.

Pada lapisan tersembunyi dengan variasi simulasi dengan jumlah unit *neuron* sebanyak 20, 30, dan 40 unit *neuron*, jumlah tersebut dihitung berdasarkan Persamaan 1 berikut ini.

$$nh = \frac{1}{2}(ni + no) + \sqrt{ndt} \quad (1)$$

dengan nh = jumlah minimal *neuron* pada *hidden layer*

ni = jumlah *neuron* input

no = jumlah *neuron* output

\sqrt{ndt} = jumlah data untuk proses pelatihan

Sedangkan pada layer output menggunakan 1 unit *neuron*.

2) Fungsi aktivasi

Fungsi aktivasi yang digunakan pada lapisan tersembunyi dan layer output adalah fungsi *logistic sigmoid*.

3) Algoritme pelatihan

Algoritme pelatihan yang digunakan dalam simulasi *trainbr*, *trainlm*, dan *trainbfgs*.

4) Bobot awal dan bias awal yang digunakan dalam jaringan ditetapkan secara acak.

5) Maksimum epochs

Maksimum *epoch* adalah jumlah *epoch* maksimal yang dilakukan selama proses pelatihan. Iterasi akan dihentikan apabila nilai *epoch* melebihi maksimum *epoch* yang telah ditentukan. Dalam penelitian ini jumlah maksimum *epoch* ditentukan sebesar 1000.

6) Laju Pembelajaran/learning rate

Dalam penelitian ini, simulasi dilakukan dengan variasi laju pembelajaran jaringan yaitu sebesar 0.05, 0.1, dan 0.5.

7) Kinerja tujuan/goal performance

Dalam penelitian ini kinerja tujuan ditentukan dengan nilai $MSE \leq 0,001$. Kemudian nilai *Mean Square Error* yang diperoleh dari proses *training* masing-masing arsitektur diamati untuk dicari nilai MSE optimal yang akan menjadi arsitektur jaringan terbaik untuk peramalan.

4. Hasil dan Pembahasan

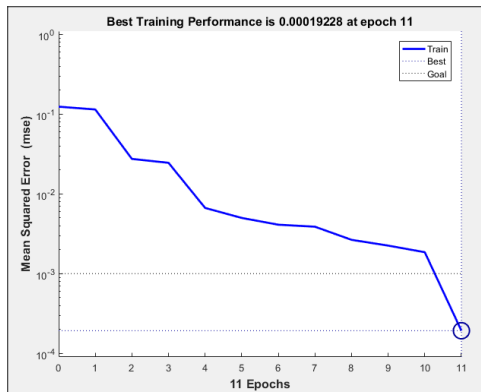
a) Pelatihan

Data yang digunakan untuk pelatihan adalah data jumlah pengguna kereta api pada periode bulan Januari 2006 sampai dengan bulan Desember 2012. Data tersebut dibagi menjadi 2 bagian yaitu data input dan data output. Data bulan Januari 2006 sampai dengan bulan November 2012 sebagai data input. Dan data bulan Januari 2007 sampai dengan bulan Desember 2012 sebagai data output/target.

Untuk membangun jaringan syaraf tiruan dengan model *feedforward backpropagation* pada perangkat lunak Matlab, digunakan perintah “*newff*”. Kemudian untuk algoritme pelatihan yang menggunakan perintah “*trainlm*” untuk algoritme *levenberg marquardt*, “*trainbr*” untuk algoritme *bayesian regularization*, dan perintah “*tranbfg*” untuk algoritme *quasi newton*.

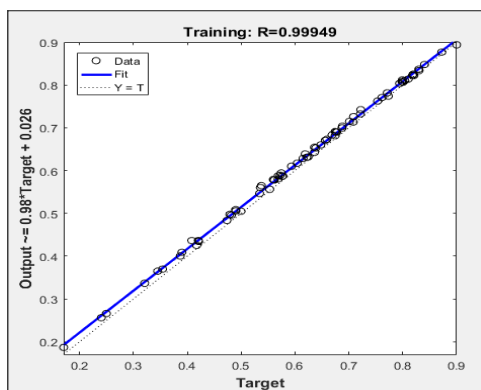
Berikut ini merupakan hasil dari simulasi dari jaringan yang dibangun :

Berdasarkan simulasi dilakukan menggunakan *trainlm levenberg marquardt*, penggunaan 30 unit neuron pada lapisan tersembunyi dengan *learning rate* 0,5 menunjukan nilai MSE yang optimal. Arsitektur jaringan ini memberikan nilai MSE *training* sebesar 0,000192 dengan tingkat konvergensi sebesar 11 iterasi/*epoch* seperti yang terlihat pada Gambar 2.



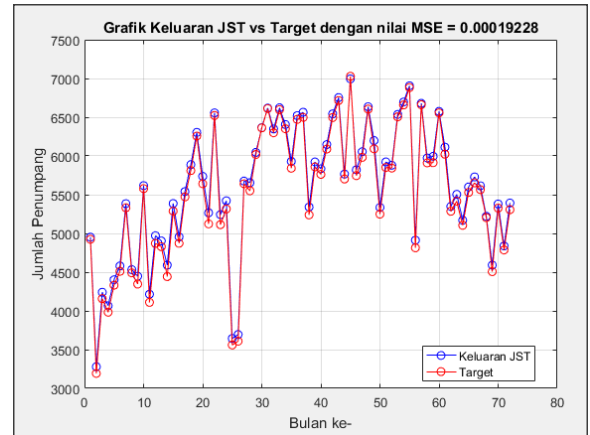
Gambar 2 MSE Pelatihan

Selain itu, ketepatan antara data *output* dengan data target juga sangat baik dengan nilai *regresi linear* mendekati 1, hal tersebut ditunjukkan pada Gambar 3.



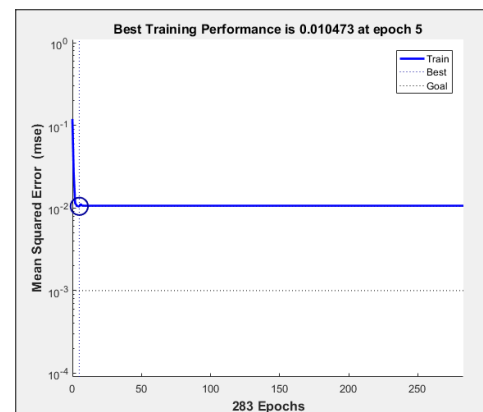
Gambar 3 Plot Regresi Linear

Apabila dibandingkan dengan nilai MSE dan *epoch* dari jaringan lain, maka dapat disimpulkan bahwa arsitektur jaringan terbaik dalam pelatihan ini adalah jaringan dengan 12 *neuron input*, 1 *hidden layer* dengan 30 unit *neuron*, dan 1 *neuron output layer* (12-30-1) yang mampu mengenali pola data dengan cukup baik seperti terlihat pada Gambar 4.



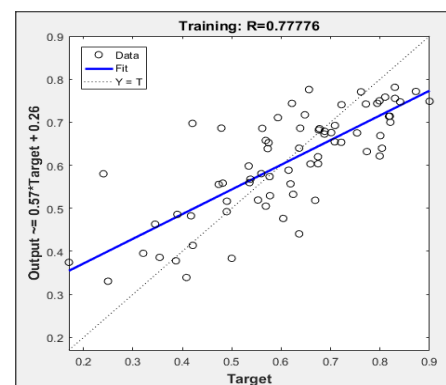
Gambar 4 Pola data aktual vs data output

Berdasarkan simulasi menggunakan *trainbr bayesian regularization*, penggunaan 40 unit *neuron* pada lapisan tersembunyi dengan *learning rate* 0,1 menunjukan nilai MSE yang optimal. Arsitektur jaringan ini memberikan nilai MSE *training* sebesar 0,010473 dengan tingkat konvergensi sebesar 283 iterasi/*epoch*, seperti yang terlihat pada Gambar 5.



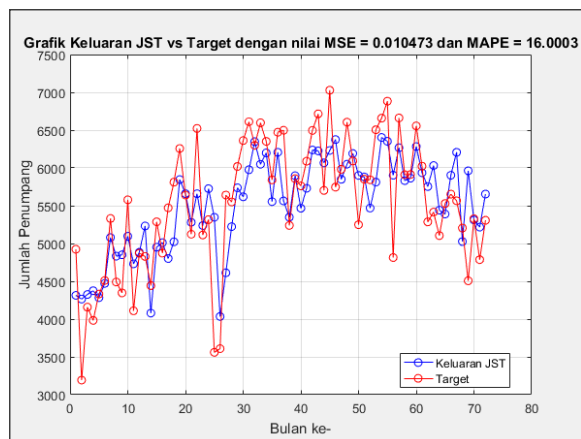
Gambar 5 MSE Pelatihan

Apabila dibandingkan dengan nilai MSE dan *epoch* dari jaringan lain, maka dapat disimpulkan bahwa arsitektur jaringan terbaik dalam pelatihan ini adalah jaringan dengan 12 *neuron input*, 1 lapisan tersembunyi dengan 40 unit *neuron*, dan 1 *neuron output layer* (12-40-1). Tetapi algoritme ini menunjukan hubungan output dan target yang kurang baik, ditunjukkan dengan nilai pada plot regresi linear pada Gambar 6.



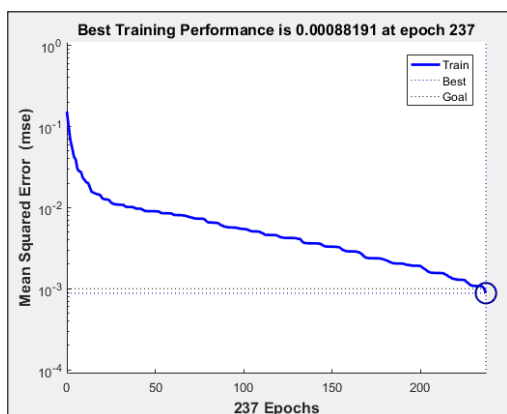
Gambar 6 Plot Regresi Linear

Dari hasil MSE dan regresi yang jauh dari target, maka dalam pengenalan pola data tidak menunjukkan hasil yang akurat seperti pada Gambar 7 berikut.



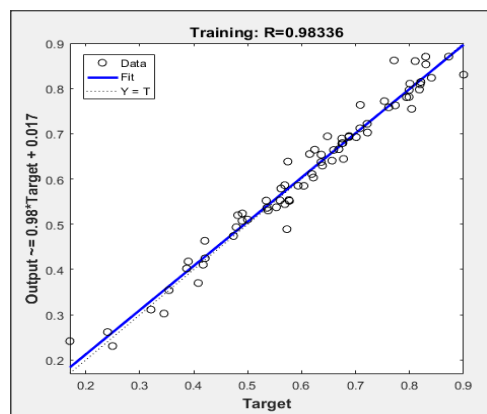
Gambar 7 Pola data aktual vs data output

Berdasarkan simulasi menggunakan *trainbfgs quasi newton*, penggunaan 30 unit *neuron* pada *hidden layer* dengan learning rate 0,01 menunjukkan nilai MSE yang optimal. Arsitektur jaringan ini memberikan nilai MSE *training* sebesar 0,000881 dengan tingkat konvergensi sebesar 237 iterasi/*epoch*, seperti terlihat pada Gambar 8.



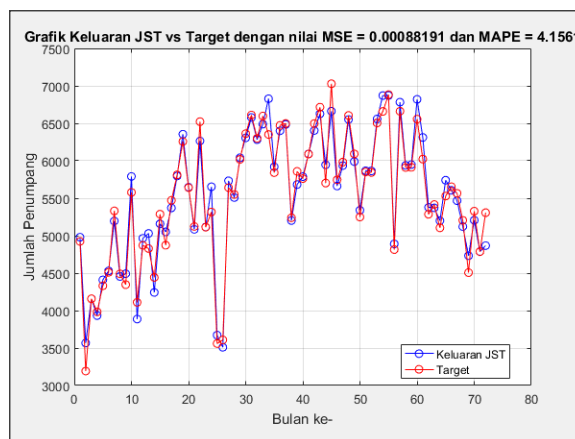
Gambar 8 MSE Pelatihan

Pada plot regresi linear pada Gambar 9 menunjukkan hasil yang baik dengan nilai mendekati 1.



Gambar 9 Plot Regresi Linear

Apabila dibandingkan dengan nilai MSE dan *epoch* dari jaringan lain, maka dapat disimpulkan bahwa arsitektur jaringan terbaik dalam pelatihan ini adalah jaringan dengan 12 *neuron input*, 1 *hidden layer* dengan 30 unit *neuron*, dan 1 *neuron output layer* (12-30-1) yang mampu mengenali pola data yang cukup akurat yang ditunjukkan pada Gambar 10 dibawah ini.



Gambar 10 Pola data aktual vs data output

Pada simulasi pelatihan yang telah dilakukan di masing-masing arsitektur dan algoritme pelatihan, hasilnya menunjukkan bahwa arsitektur 12-30-1 dengan algoritme *levenberg-marquardt* memiliki tingkat MSE terkecil dengan nilai 0,000192. Tabel 1 berikut ini menampilkan hasil dari simulasi tersebut.

Tabel 1 Perbandingan Hasil Simulasi

No	Neuron hidden layer	Target	Algoritme Pelatihan		
			<i>trainbr</i>	<i>trainlm</i>	<i>trainbfgs</i>
1	20	0.001	0.010601	0.000761	0.000996
2			0.010625	0.000904	0.000988
3			0.010622	0.000942	0.000926
4	30	0.001	0.010632	0.000699	0.000881
5			0.010561	0.000925	0.000959
6			0.010632	0.000192	0.000999
7	40	0.001	0.010633	0.000782	0.000994
8			0.010473	0.000988	0.000996
9			0.010484	0.000965	0.000959

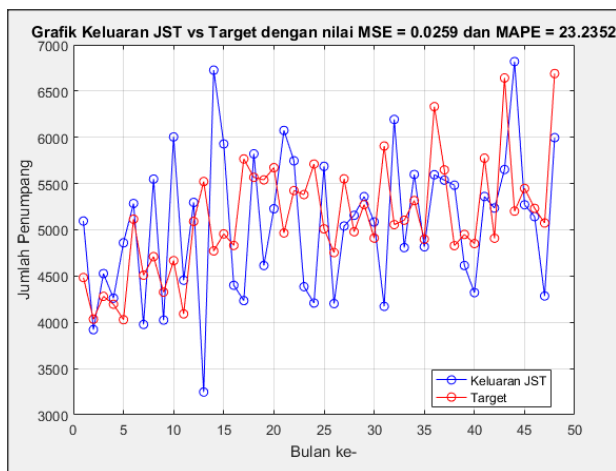
b) Pengujian

Setelah dilakukan simulasi pada proses *training*, selanjutnya dilakukan proses *testing* untuk menggeneralisasi hasil yang diperoleh pada proses *training* untuk mengetahui apakah arsitektur jaringan yang diperoleh sudah layak untuk digunakan. Proses *testing* memiliki prosedur simulasi yang sama dengan proses *training*, namun data yang digunakan adalah data periode bulan Januari 2012 sampai dengan bulan desember 2016. Yaitu data bulan Januari 2012 sampai

bulan November 2016 sebagai input dan data dari bulan Januari 2013 sampai bulan Desember 2016 sebagai output.

Hasilnya menunjukkan nilai MAPE dari jaringan tersebut sebesar 23%. Berdasarkan kriteria MAPE, rentang persentase suatu jaringan dikatakan cukup apabila nilai MAPE berada pada nilai 20% - 50% [7]. Hal tersebut menunjukkan bahwa jaringan yang dibangun dengan arsitektur 12-30-1 dengan fungsi pelatihan *levenberg-marquardt* ini dinilai cukup untuk digunakan.

Berikut Gambar 11 yang menunjukkan hasil simulasi pengujian pengenalan pola data yang menggunakan arsitektur jaringan optimal yang diperoleh simulasi algoritme pelatihan *levenberg marquardt*.



Gambar 11 Hasil pengujian algoritme *trainlm*

5. Kesimpulan

Pada proses pelatihan, model *feedforward backpropagation neural network* dengan algoritme pelatihan *levenberg marquardt* menunjukkan arsitektur optimal 12-30-1 dengan MSE terkecil dengan nilai 0,000192 dengan laju pembelajaran 0,5.

Pada proses pengujian, arsitektur optimal 12-30-1 dengan algoritme pelatihan *levenberg marquardt* menghasilkan MSE sebesar 0,0259 dan nilai MAPE menunjukkan nilai sebesar 23%. Hal tersebut berarti bahwa jaringan yang dibangun dengan arsitektur 12-30-1 dengan fungsi pelatihan *levenberg-marquardt* ini dinilai cukup untuk digunakan.

Daftar Pustaka

- [1] "Laporan Tahunan Pt. KAI Persero Tahun 2015, www.kip.kereta-api.co.id."
- [2] K.-T. Park and J.-G. Baek, "Various Type of Wavelet Filters on Time Series Forecasting," 2017, pp. 258–259.
- [3] A. Azadeh, S. F. Ghadrei, and B. P. Nokhandan, "One day-ahead price forecasting for electricity market of Iran using combined time series and neural network model," in *Hybrid Intelligent Models and Applications, 2009. HIMA'09. IEEE Workshop on*, 2009, pp. 44–47.
- [4] P. Xin and H. Zhao, "Time series forecasting using multilayer neural network constructed by a Monte-Carlo based algorithm," in *Web Society, 2009. SWS'09. 1st IEEE Symposium on*, 2009, pp. 264–267.

- [5] F. Azmi, "Analisis Leaning Jaringan RBF (Radial Basis FUnction Network) Pada Pengenalan Pola Alfanumerik," *J. TIMES*, vol. 5, no. 2, pp. 32–34, 2017.
- [6] C. Benhamed, S. Mekaoui, and K. Ghomid, "A large scale IP Network Traffic matrix Estimation based on ANN: A comparison study on training algorithms," in *Electrical Engineering (ICEE), 2015 4th International Conference on*, 2015, pp. 1–6.
- [7] P. Nugroho and others, "Model Peramalan Jumlah Pernikahan Menggunakan Metode Fuzzy Time Series dengan Algoritma Average Based Length pada KUA," *HIMSYATECH*, vol. 10, no. 2, 2014.